

TARTU ÜLIKOOL  
LOODUS- JA TEHNOLOOGIATEADUSKOND

Füüsika instituut

Kait Krull

**LISAMÄLU ÜHENDUSVÕIMALUSE LOOMINE  
MINDSTORMS PLATVORMILE**

Bakalaureusetöö (12 EAP)

Juhendaja: Heilo Altin, MSc

Kaitsmisele lubatud .....

Juhendaja .....

*allkiri, kuupäev*

Tartu 2013

# Sisukord

Kasutatud lühendid ja definitsioonid .....	4
Sissejuhatus .....	7
1. Ülevaade.....	9
1.1 Kooliroboti projekt .....	9
1.2 LEGO MINDSTORMS algusaastad .....	9
1.3 LEGO MINDSTORMS NXT .....	9
1.4 LEGO MINDSTORMS NXT juhtplokk .....	10
1.5 LEGO MINDSTORMS NXT mälu .....	11
1.6 Ülevaade SD mälukaardist .....	12
1.6.1 SD mälukaardi protokollid.....	14
1.6.1.1 SD mälukaardi SD režiim.....	14
1.6.1.1.1 SD 1-bitine protokoll.....	14
1.6.1.1.2 SD 4-bitine protokoll.....	14
1.6.1.2 SD SPI režiim .....	15
1.6.2 SD käsud .....	15
1.7 FAT failisüsteem .....	16
1.8 Ühendusvõimaluse leidmine .....	17
1.9 LEGO MINDSTORMS NXT Bluetooth.....	18
1.9.1 Bluetooth ülema-alluva-süsteem.....	19
1.9.2 Mälumooduli Bluetooth moodul.....	20
2. Tulemus.....	21
2.1 Mälumoodul .....	21
2.1.1 Elektroonika skeem.....	21
2.1.2 Füüsiline ehitus .....	23
2.1.3 Programmikood .....	25
2.1.3.1 Peaprogramm.....	25
2.1.3.2 SD kaart.....	26
2.1.3.3 RTC .....	27
2.1.3.4 Bluetooth .....	28
2.2 Programmeerimisplukkide loomine NXT-G'le.....	28

2.2.1	BTMemory plokk .....	28
2.2.2	ÜlemVI .....	29
2.2.3	Konfiguratsiooni VI.....	29
2.2.4	Joonistamise VI.....	31
2.2.5	AlamVI .....	32
2.3	Mälumooduli tarkvara silumine, testimine ning tulemused .....	32
2.3.1	Testimine arvuti ja terminal tarkvaraga .....	32
2.3.2	Testimine NXT juhtplokiga .....	33
2.3.3	Bluetooth ühenduse testimine.....	33
2.3.4	Mälumooduli voolutarbimine .....	34
2.3.5	Testimine erinevate kaartidega .....	35
	Kokkuvõte .....	36
	Creating an external data drive for MINDSTORMS robotics platform .....	37
	Summary .....	37
	Kasutatud allikad.....	38
	LISAD .....	40
	LISA 1 – CD.....	40
	Lisa 2 – Elektroonika skeem .....	41
	LISA 3 – BTMemory ploki paigaldamine NXT-G 2.0 tarkvarasse. ....	42
	LISA 4 – Ploki kasutusjuhend.....	43

## Kasutatud lühendid ja definitsioonid

FLASH (ingl k <i>Flash memory</i> )	Välmälu, elektrilise välgkustutusega säilmälu, milles andmeid adresseeritakse baidikaupa, kirjutatakse ja kustutakse plokkidena.
ASCII (ingl k <i>American Standard Code of Information Interchange</i> )	infovahetuse standardkood tähtede, numbrite ning talitlusmärkide kodeerimiseks 7 bitiga. Laiendatud süsteemis 8 bitiga.
Boodikiirus (ingl k <i>Baud Rate</i> )	Andmeedastuskiirus, edastatakse bitt sekundis (b/s).
CMOS (ingl k <i>Complementary metal oxide semiconductor</i> )	Komplementaarne metall-oksiid-pooljuht-struktuur.
CRC (ingl k <i>Cyclic Redundancy Check</i> )	Tsükkelkoodkontroll. Meetod üle sideliini edastatud andmete tervikluse kontrolliks.
Diskreetimissagedus, näiduvõtusagedus (ingl k <i>sample rate</i> )	Analoogsignaali digiteerimisel signalist sekundis võetavate näitude arv.
ECC (ingl k <i>Error Correction Code</i> )	Veaparanduskood, erineb paarsus-kontrolli meetodist selle poolest, et vigu mitte ainult ei otsita, vaid ka parandatakse.
FAT (ingl k <i>File Allocation Table</i> )	Failipaigutustabel, sisaldab failiosade aadresse, mis määravad nende asukohad nt kõvaketta klastrites, tabelis FAT32 võib klatri maht olla 512 baitikuni 32 kilobaiti.

I <sup>2</sup> C (ingl k <i>Inter-Integrated Circuit</i> )	Kahejuhtme liides, kasutatakse madalama andmevahetuskiirusega seadmete ühendamiseks.
Klaster (ingl k <i>cluster</i> )	Kettal korraga hallatav loogikaline mäluüksus.
LDO (ingl k <i>Low Dropout</i> )	Madal pingelang.
LED (ingl k <i>Light Emitting Diode</i> )	Valgusdiod. Pooljuhtdiod, mis läbiva voolu toimet kiirgab valgust.
Liides (ingl k <i>Interface</i> )	Arvutisüsteemis andmevahetuseks vajalike omadustega liitekoht tarkvara ja riistvara üksuste kujul.
LSB (ingl k <i>Least Significant Bit</i> )	Madalaim bitt.
MCU (ingl k <i>Microcontroller Unit</i> )	Mikrokontroller.
MMC (ingl k <i>Multimedia Card</i> )	MMC-mälukaart.
MSB (ingl k <i>Most Significant Bit</i> )	Tähtsaim bitt.
Port (ingl k <i>Port</i> )	Sisenemise ja väljumise koht.
Protokoll (ingl k <i>Protocol</i> )	Määrab funktsionaalüksuste vahelise suhtluse korra (toimingud, vormingud) teatud valdkonnas.
<i>Pull-up</i> takisti	<i>Pull-up</i> takisti eesmärk on ära hoida sisendi „ujumine“ staatilise elektri ja muude häirete tõttu.
Püsivara (ingl k <i>Firmware</i> )	Püsimälu salvestatud juhtimisprogramm.
RAM (ingl k <i>Random Access Memory</i> )	Suvaline mälu, mille suvalise mälupesa poole saab andmebaitide kirjutamiseks ja lugemiseks otse

	pöörduda.
RTC (ingl k <i>Real Time Clock</i> )	Reaalajakell, mis töötab patarei toitet, kui seade on võrgutoitest lahutatud.
Sektor (ingl k <i>Sector</i> )	Ketta vähim füüsiliselt adresseeritav salvestusühik.
SPI (ingl k <i>Serial Peripheral Interface</i> )	Neljajuhtme sünkroonne jadaliides, mida kasutatakse seadmete ühendamiseks. Võimalik samaaegne andmete saatmine ja vastuvõtmine.
TWI (ingl k <i>Two Wire Interface</i> )	Kahejuhtme jadaliides, mida kasutab Atmel. Sarnaneb I <sup>2</sup> C-ga.
UART (ingl k <i>Universal Asynchronous Receicer/Transmitteri</i> )	Universaalne asünkroonne saatja-vastuvõtja. Asünkroonne ühepolaarne andmevahetus, võimalik on samaaegne saatmine ning vastuvõtmine.
VI (ingl k <i>Virtual Instrument</i> )	LabVIEW programmifail, mis koosneb kolmest osast: plokkiagrammist, esipaneelist ning konnektorist.
Ülem-alluv-süsteem (ingl k <i>Master-Slave-System</i> )	Andmesidesüsteem, milles algatusvõime on ainult ühel komponendil ehk ülemal (ingl k <i>master</i> ), mis algatab ja juhib sideseansse. Alluv (ingl k <i>slave</i> ) reageerib ülema käskudele.
Numbrisüsteemid	Töös on kahendsüsteem on kujutatud „“ („1001“) ning kuueteistkümnendsüsteem on kujutatud eesliitega 0x (0x05).

## Sissejuhatus

Üldhariduskoolides on viimastel aastatel edendatud projekte, mille abil soovitakse parandada õppekvaliteeti ja muuta reaalsed atraktiivsemateks. Kaasates õpilasi lahendama ülesandeid praktiliselt, on muutunud ainete teoreetilise osa õpilastele paremini vastuvõetavamaks.

Koolide õppekavades leidub mitmeid teemasid, mille mõistmisel on abiks mõni rakenduslik näide. Projektid, nagu näiteks Kooliroboti projekt ning TERECoP<sup>[4]</sup>, aitavad koolidel muretseda robotika komplekte, mille abil tutvustatakse õppeainetes seostuvaid teemasid ning probleemide lahendamise olemust.

Eestis tegutsev Kooliroboti projekt kasutab õppevahenditena peamiselt LEGO, mis on koostöös MIT Meedia Laboriga<sup>[16]</sup> loonud seadmete seeria MINDSTORMS.<sup>[5]</sup>

LEGO MINDSTORMS seeria tähtsaim element on programmeeritav juhtplokk. Selle abil saavad õpilased luua projekte, mis võimaldavad rakendada õpitud teooriat praktiliselt. Lisaks kuuluvad komplekti heli-, kaugus-, puute- ja värviandurid, mootorid, kaablid, ehitusosad ning lisatarvikud vastavalt versioonile. Kokku on kaks peamist juhtplokki: RCX ning NXT.<sup>[5]</sup>

Bakalaureusetöös keskendutakse LEGO MINDSTORMS NXT platvormile, mis nüüdseks on asendanud vanema põlvkonna seeria, LEGO MINDSTORMS RCX.

NXT juhtplokil on kindel mäluruum, mida ei saa suurendada. Kasutajal on võimalus luua programme ning faile olemasoleva mäluruumi arvelt, mida kohati jääb liiga väheks. Ruumi puudusel pakutakse lahenduseks helifailide, näiteprogrammide ning mittevajalike failide kasutajapoolne eemaldamine. Tegemist ei ole aga lahendusega vaid viisiga probleemi vältida.

MINDSTORMS-il on lisaks kaasasolevatele anduritele võimalus juurde osta kolmandate osapoolte andureid, millega saab teostada mitmesuguseid mõõtmisi. Mõõtmiste andmeid saab koguda, kuid selleks on vajalik vaba mälu, mida hetkel on juhtplokil vähe. Parim lahendus probleemile oleks välja töötada riistvaraline moodul, lisamälu, mis suhtleks NXT juhtplokiga ning oleks võimeline sinna andmeid salvestama.

Käesoleva bakalaureusetöö eesmärgiks on väljatöötada lahendus LEGO MINDSTORMS NXT mälu vähesuse probleemile. Selleks leitakse ühendusvõimalus, mille alusel hakatakse disainima mälumoodulit. Lisaks dokumenteeritakse töö protsessi piisavalt, et seda oleks võimalik kasutada teiste laiendusmoodulite loomisel. Loodava mälumooduli arendamisega luuakse kasutatav programmeerimisplok NXT-G tarkvarale koos juhendiga, püsivara mälumoodulile ning riistvaraline lahendus.

Bakalaureusetöö on jagatud kaheks suuremaks osaks, kus kirjeldatakse järgnevat:

- Esimeses osas tutvustatakse üldiselt LEGO MINDSTORMS platvormi ja Eestis koolide robotikaga tegelevat Kooliroboti projekti. Analüüsitakse kasutatavat platvormi ning leitakse ühendusviis, läbi mille hakatakse disainima mälumoodulit. Kirjeldatakse FAT failisüsteemi ning välismäluna kasutatavat SD mälukaarti.
- Teises osas kirjeldatakse mälumooduli loomise tööprotsessi ning vajalikke lisavahendeid. Luuakse programmikood mälumoodulile ning NXT-G tarkvaraplokid. Lisaks tuuakse välja loodud mälumooduli testimise protsess ja tulemused.



# 1. Ülevaade

## 1.1 Kooliroboti projekt

Kooliroboti projekt loodi 2007. aasta kevadel. Projektiga on hetkel liitunud 110 kooli (seisuga 16.11.2012). Selle raames korraldatakse õpetajatele koolitusi ja tagatakse õppematerjalide olemasolu.

Koolide ülesanne on leida sponsorid või esitada taotlus toetuse saamiseks SA Tiigrihüppele vajalike robotikakomplektide otsmiseks. Projekti läbiviimiseks kasutatakse peamiselt LEGO MINDSTORMS NXT seadmeid.<sup>[10]</sup>

## 1.2 LEGO MINDSTORMS algusaastad

LEGO MINDSTORMS toodi esmakordselt turule 1998. aastal. Sel hetkel kandis komplekt nime *LEGO MINDSTORMS Robotic Invention Kit* ja sisaldas 717 osa, kuhu kuulusid erinevad andurid, mootorid, käigud, ehitusklotsid ning RCX juhtplokk (ingl k *RCX Brick*).

Tootesarja tegi eriliseks mikrokontrolleriga juhtplokk, mida sai programmeerida. See omadus võimaldas luua uudseid projekte, mida enne ei olnud võimalik saavutada. Toodet müüdi vähem kui kolme kuuga üle 80 000 eksemplari. Komplekti hind oli tol hetkel \$ 200 ja kogu müügiga tõsteti firma toodangu läbimüüki aastaga ligi 300 %.

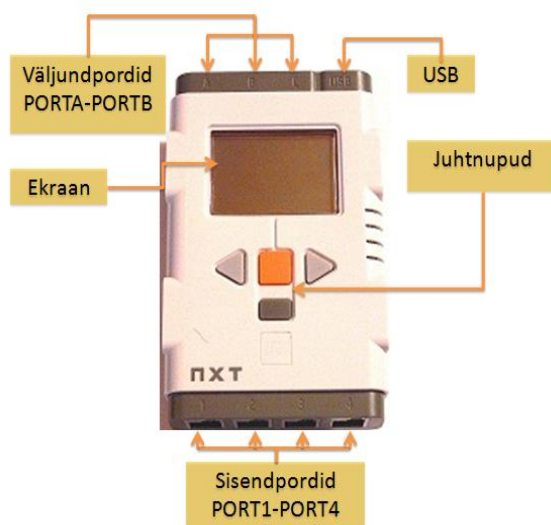
MINDSTORMS idee arendati välja MIT Meedia Laboris (ingl k *MIT Media Lab*) projektina, mida juhtisid Seymour Papert ja Mitchel Resnick. MINDSTORMS nimi pärineb Seymour Papert'i 1980. a. teosest, kus ta kirjeldas, et programmeerimine võibki olla parim viis õpetada lastele probleemide lahendamise olemust. Hiljem rahastas LEGO projekti, millest varsti sündiski LEGO MINDSTORMS seeria.<sup>[5]</sup>

## 1.3 LEGO MINDSTORMS NXT

LEGO MINDSTORMS NXT toodi turule 2006. aasta juulis. Komplekt loodi eesmärgiga, et see oleks kümneaastasele lapsele piisavalt lihtne monteerida ning programmeerida umbes 30 minutiga. Kuna NXT juhtplokiga saab luua väga mitmekülgseid ja õpetlikke projekte, kasutatakse seda nii kesk- kui ülikoolides.

NXT seeriast on turule toodud mitu versiooni: NXT 1.0 (juuli 2006), NXT 2.0 (august 2009) ja NXT *Educational Version* (õppeversioon). Igas komplektis on NXT juhtplokk, LEGO Technics ehitusklotsid, andurid, mootorid ja vastavalt komplektile veel mõned lisakomponendid. Kõik versioonid kasutavad sama juhtplokki.<sup>[1]</sup>

## 1.4 LEGO MINDSTORMS NXT juhtplokk



**Joonis 1. LEGO MINDSTORMS NXT juhtplokk.**

NXT juhtplokk on komplekti kõige tähtsam osa. (Joonis 1) Tegemist on programmeeritava seadmega, mida juhitakse kahe Atmeli mikrokontrolleriga. Peamiseks mikrokontrolleriks on 32-bitine ARM7 (AT91SAM7S256), mis töötab 48 MHz taktsagedusega. Sellel on 256 KiB suurune FLASH mälu ning 64 KiB suurune RAM. Teine mikrokontroller on 8-bitine ATmega48, mis töötab 8 MHz taktsagedusega. Sellel mikrokontrolleril on 4 KiB suurune FLASH mälu ning 512 B RAM.

Seadmel on 100 x 64 piksliga must-valge LCD maatriksekraan (26 x 40,6 mm). Ekraani kasutatakse mitmesuguse informatsiooni kuvamiseks. Peamiselt kuvatakse sellele püsivara kasutajaliides, informatsioon loodud programmi kohta või programmi väljundid.

Juhtplokki sees on valjuhääldi, mis annab kasutajale märku seadme töötamisest, nupule vajutamisest, programmi allalaadimisest. Valjuhääldi on võimeline ette mängima helisid, mille näiduvõtusagedus ehk diskreetimissagedus on 2 - 16 kHz. Toetatud on vaid 8-bitised .rso tüüpi helifailid.

Kasutamise lihtsustamiseks on seadme esipaneelil neli nuppu (Joonis 1), millega liigutakse ringi kasutajaliidese menüüs. Lisaks saab neid kasutada programmides ja seadme sisse- ja väljalülitamiseks.

Suhtlus arvutiga toimub läbi USB – NXT juhtplokil on selleks eraldi port. (Joonis 1) Selle kaudu uuendatakse seadme püsivara, laaditakse alla programme ning seda kasutatakse andmete lugemiseks ja muutmiseks. Lisaks toetab juhtplokk ka Bluetooth 2.0 tehnoloogiat, mis lubab andmeid saata ning vastu raadioside abil.

Juhtplokil on neli sisendporti (Joonis 1), mis kasutavad ühenduseks kuuetraadilist kaablit, mis võtavad vastu nii digitaal- kui analoogsignaale. Sisendpordid on tähistatud numbritega 1-4. Port 4 toetab ainsana kahte protokollit. Kiiremate ühenduste loomisel kasutatakse RS-485 protokollit, mille kiirus on 921,6 kb/s, muul juhul kasutavad sisendid I<sup>2</sup>C suhtlusprotokollit, mille kiirus on 9,6 kb/s.

Protokollid kasutavad ülem-alluv-süsteemi (*master-slave*), kus NXT täidab alati *master* ehk ülema funktsiooni, tähendades ühtlasi, et andmevoogu kontrollib NXT.

Lisaks sisendportidele on juhtplokil kolm väljundporti (Joonis 1). Neid tähistatakse tähtedega A, B ja C. Ühenduseks kasutatakse kuuetraadilist kaablit, millega saadetakse informatsiooni väljundile ning vajadusel ka loetakse. Neid porte kasutatakse mootorite juhtimiseks.

NXT toiteallikaks saab kasutada kuute AA 1,5 V patareid või laetavat 7,4 V liitiumioon akut mahutavusega 1400 mAh.<sup>[13]</sup>

## **1.5 LEGO MINDSTORMS NXT mälu**

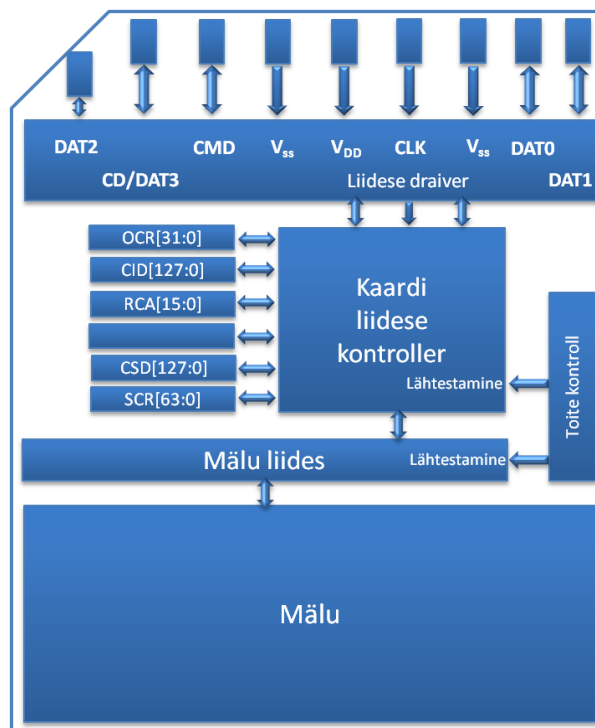
Juhtplokil on piiratud mälumaht, mida kokku on 256 KiB. Mäluna kasutatakse Atmeli ARM7 (AT91SAM7S256) mikrokontrolleri väikmälu. Sellesse on paigaldatud seadme püsivara ning kasutatavad failid.

Mälu on jagatud 1024 leheküljeks, millest iga lehekülg suudab hallata maksimaalselt 256 baiti. Ametlik püsivara kasutab umbes 125 KiB mälu, jättes kasutajale vähem kui 130 KiB. Hulgani leiub väiksemaid püsivarasid nagu leJOS NXJ<sup>[20]</sup>, mis vajavad eraldi paigaldamist ning võimaldavad kasutada teist programmeerimiskeelt ning -keskkonda.

Programmi jooksutatakse välmälust, kuid tähtsamad funktsioonid laaditakse suvapöördusmällu (RAM). Suvapöördusmälu kasutatakse, kuna välmälu ei ole võimalik samaaegselt lugeda ning kirjutada.<sup>[1]</sup>

## 1.6 Ülevaade SD mälukaardist

Bakalaureusetöö eesmärgiks oli leida lahendus informatsiooni talletamiseks. Selleks sai valitud *Secure Digital Card* ehk SD mälukaart, kuna tegemist on üha rohkem kasutatava andmekandajaga ning suure kasutuse tõttu levib kaarte tihti igas vähegi välist mälu kasutavas seadmes.



**Joonis 2. SD mälukaart.**

Joonisel 2 on toodud skeem SD mälukaardi ehitusest. Kaardis on eraldi kontrolleri, mis korraldab andmevahetust FLASH mälu tüüpi moodulite vahel. Kontrolleri saadetakse käsked ning andmeid. Vastavalt käskule teostatakse erinevaid ülesandeid.

SD standardsuurusega kaardi mõõtmed on 32 x 24 mm, mis on samad nüüdseks üha vähem kasutatust leidva MMC kaardiga. Kahe kaardi erinevus on nende väljaviikude arvu hulk, paiknemine (Tabel 1) ning ka paksus. SD kaardi paksus on 2,1 mm, kui MMC paksus on 1,4 mm. SD kaart oli algselt projekteeritud krüpteeritud andmete

talletamiseks, kuid suhteliselt suurele andmeedastuskiirusele kasutakse neid laialdaselt krüpteerimata andmete salvestusseadmena.

**Tabel 1. MMC ja SD mälukaartide väljaviikude erinevused.**

Nr	SD	MMC	SPI
8	DAT1		
7	DAT0	DAT	DO
6		Vss2	
5	CLK	CLK	SCLK
4		Vcc	
3		Vss1	
2	CMD	CMD	DI
1	CAT3	RES	CS
9	DAT2		

Tänapäeval on kokku kolm peamist SD kaardi suurust: SD, miniSD ja microSD. Kõiki vastavaid suuruseid saab kasutada ühendusadapteriga, muutes väiksemat vormi suuremaks. Lisaks on laiendatud kiirusklassid 2, 4, 6 ja 10, mis tähistavad minimaalset kirjutamiskiirust MB/s. Laiendusena on veel kasutusel kaartide lahterdamine erinevate pakketehnoloogiate järgi: SDHC, SDXC ja SDIO.

Selleks, et tänapäeva operatsioonisüsteemid oskaksid talletatud informatsiooni kasutada, peavad SD mälukaardid olema vormindatud failisüsteemi. Selleks on vajalik tagada ka loodaval mälumoodulil failisüsteemi tugi. Kasutatavaks failisüsteemiks on valitud FAT32, mida toetavad kõik enamlevinumad operatsioonisüsteemid. Tavaliselt on väiksema andmemahuga SD mälukaardid vormindatud FAT16 failisüsteemi, kuid nende vormindamine FAT32 failisüsteemi võimaldab kasutada väiksemaid klastreid, mis lubavad väikeste failide puhul efektiivsemat mälukasutust.

SD kaardid ei tarbi palju voolu. Andmelehe kohaselt tarbib aktiivses olekus olev SD mälukaart maksimaalselt 100 mA. Tänu väikesele voolutarbele sobib SD kaart hästi mäluseadme välismäluks. Kaardi kasutamiseks on vaja tagada toitepinge vahemikus 2,7 V – 3,6 V, mille väärtusi võrreldakse OCR registri väärtustega. Soovituslik oleks kasutada 3,3 V toitepinget. Seadme mittetöötamise põhjuseks kaardi sisestamisel võib olla suurema andmeedastuskiirusega kaardi kasutamine, mis vajab suuremat pinget kui seade pakub.<sup>[18]</sup>

### 1.6.1 SD mälukaardi protokollid

Järgnevalt on toodud SD mälukaardi väljaviikude tabel, millelt on näha, et osadel viikudel on mitu kasutuseesmärki. SD kaart toetab kokku kolme andmeedastusprotokolli. Allpool toodud tabelis on kirjeldatud SD režiimi ning SPI režiimi. (Tabel 2) SD režiimi kuulub kaks protokoll: 1-bitine - ning 4-bitine protokoll.

**Tabel 2. SD- ja SPI režiim.**

Viik	Viigu nimi	Funktsioon (SD režiim)	Funktsioon (SPI režiim)
1	DAT3/CS	<i>Data Line 3</i> Andmeliin 3	<i>Chip Select/Slave Select (SS)</i> Kiibi valik/alama valik
2	CMD/DI	<i>Command Line</i> Käsu liin	<i>Master Out Slave In (MOSI)</i> Ülem välja alam sisse
3	VSS1	<i>Ground</i> Maa	<i>Ground</i> Maa
4	VDD	<i>Supply Voltage</i> Toitepinge	<i>Supply Voltage</i> Toitepinge
5	CLK	<i>Clock</i> Kellatakt	<i>Clock (SCK)</i> Kellatakt
6	VSS2	<i>Ground</i> Maa	<i>Ground</i> Maa
7	DAT0/DO	<i>Data Line 0</i> Andmeliin 0	<i>Master In Slave Out (MISO)</i> Ülem sisse alam välja
8	DAT1/IRQ	<i>Data Line 1</i> Andmeliin 1	<i>Unused or IRQ</i> Kasutamata viik või IRQ
9	DAT2/NC	<i>Data Line 2</i> Andmeliin 2	<i>Unused</i> Kasutamata viik

#### 1.6.1.1 SD mälukaardi SD režiim

##### 1.6.1.1.1 SD 1-bitine protokoll

SD kaardi 1-bitine protokoll on sünkroonne jadaprotokoll, kus kasutatakse ainult ühte andmeliini. Andmed edastatakse pakendamata kujul. Liini kellatakti (CLK) kasutatakse sünkroniseerimiseks ja käsuliini (CMD) käsukaadrite saatmiseks. Vastav protokoll lubab andmeliini jagamist. (Tabel 2) Lihtne ühe-ülema süsteem lubab mitmel SD kaardil kasutada sama kellatakti ja DAT0 andmeliini.<sup>[7]</sup>

##### 1.6.1.1.2 SD 4-bitine protokoll

SD kaardi 4-bitine protokoll on peaaegu samaväärne 1-bitise protokolliga, kuid põhilised erinevused on andmeliini laius, kus 1 andmeliini asemel kasutatakse 4 andmeliini. Õige disaini puhul on võimalik andmete edastuskiirust suurendada nelja kordselt võrreldes 1-bitise protokolliga. 1-biti kui ka 4-biti protokollid vajavad vaikumisi CRC kaitsekontrolli. CRC (*Cyclic Redundancy Check*) on lihtne meetod, et kontrollida

saadetud andmeplokis lihtsamaid bitivigu, mida tekitab müra andmeliinidel. 4-bitine protokoll kasutab SD kaardi nelja andmeliini (DAT), millel igaühel arvutatakse CRC 16-bitine kontrollväärtus.<sup>[7]</sup>

### 1.6.1.2 SD SPI režiim

Kolmandaks SD kaardi režiimiks on SPI (*Serial Peripheral Interface*), mille tugi on olemas paljudel mikrokontrolleritel. Tegemist on tuntud ja dokumenteeritud andmeedastussüüsiiniga, mis kasutab sünkroonset jadaprotokolli. Eeltoodud tabelis (Tabel 2) on kirjeldatud ühendatavad viigud mikrokontrolleri ja SD kaardi vahel.<sup>[7]</sup>

### 1.6.2 SD käsud

SD protokoll on käsk-vastus tüüpi (*command-response*). Kõik käsud SD kaardile saadab ülem, millele SD kaardi kontrollur saadab vastuse kaadri. SD käsud on jagatud kaheks: „CMDXX“ ja „ACMDXX“. Tegemist on vastavalt üldiste käskudega ja rakendus-spetsiifiliste (*application-specific*) käskudega, milles XX tähendab käsu numbrit.

Järgnevalt on toodud SD kaardi käsu struktuur (Tabel 3). Käsukaader koosneb kuuebaidisest struktuurist, mis saadetakse MOSI andmeliini kaudu mikrokontrollerisse. Saatmisel saadetakse esmalt MSB (*Most Significant Bit*). Esimene bait algab alati bittidega „01“, millele järgneb kuuebitine käsunumber. Järgnevad neli baiti on argument. Viimane bait on reserveeritud CRC jaoks, mis koosneb seitsmest bitist ja baidi lõppu kuuluvast lõputunnusest „1“. Vaikimisi on CRC kontrollimine SPI režiimis eemaldatud, kuid selle võib valikuliselt lisada.

**Tabel 3. SD kaardi käsu struktuur.**

Esimene bait			Baidid 2-5	Viimane bait	
0	1	Käsk	Argument (MSB enne)	CRC	1

Igale SD kaardi käsule vastatakse vastuse kaadriga. Kokku on SPI režiimis neid defineeritud kolm: R1, R2 ja R3. Igale käsule on oodatud tagastustüüp, mille abil saab kontrollida, kas vastav käsk on vastuvõetud.

SD kaardilt plokki lugemisel ja kirjutamisel lisatakse igale plokki liigutamisele algusesse algus tingimus – bait väärtusega „11111110“. Sellele järgneb plokk informatsiooni, mis on tavaliselt 512 baidi suurune. Lisaks saadetakse järgnevalt kontrolliks 16-bitine CRC, mis vaikimisi loetakse, kuid ei arvestata.

Peale igat kaardile kirjutamist, saadab SD kaart ühe baidi suuruse vastuse „XXX0AAA1“, kus XXX tähistab bitt, mida ei arvestata ja AAA tähistab seisundit. Kokku on kolm seisundsõnumit: „010“ ehk andmed võeti vastu, „101“ ehk andmeid ei võetud vastu CRC vea tõttu ja „110“ ehk andmeid ei võetud vastu kirjutamisvea tõttu.

Kaardilt lugemisel vea tekkimisel saadab SD ühe baidi suuruse veakoodi. Baidis tähistab iga bitt veakoodi, mis vastavalt positsioonile on: 0 – tähistamata viga, 1- kaardi kontrolleri viga, 2 – kaardi ECC (*Error Correction Code*) viga, 3 – ületäitumine, 4 – kaart on lukustatud, 5-7 – alati 0.

Järgnevalt on toodud SD mälukaardi põhikäsud, et seadistada kaarti ning teostada kirjutamist ja lugemist (Tabel 4).<sup>[18]</sup>

**Tabel 4. SD kaardi põhikäsud.**

Käsk	Argument	Tüüp	Kirjeldus
CMD0	Puudub	R1	Algsättestamine ning jõudeolek ( <i>idle</i> )
CMD16	32-bitine ploki pikkus	R1	Ploki pikkus
CMD17	32-bitine ploki aadress	R1	Loe plokk
CMD24	32-bitine ploki aadress	R1	Kirjuta plokk
CMD55	Puudub	R1	Järgnev käsk on rakendus-spetsiifiline (ACMDXX)
CMD58	Puudub	R3	Loe OCR ( <i>Operating Conditions Register</i> ) toitepinge kontrollimiseks
ACMD41	Puudub	R1	Initsialiseeri kaart

## 1.7 FAT failisüsteem

FAT (*File Allocation Table*) ehk failipaigutustabel on failisüsteem. Selles kirjeldatakse failide füüsilist paigutust andmekandjal. Failid paigutatakse klastritesse, kus iga fail kasutab minimaalselt ühte klastrit. Vastav tabel asub kaitstud mälupiirkonnas.

Klastrid (loogikalised üksused) koosnevad fikseeritud suurusega sektoritest (füüsilised üksused) ja on adresseeritud n-bitiste kannetega aadressiruumi (tabelisse), kus n on sõltuvalt FATi versioonist 12 (FAT12), 16 (FAT16) või 32 bitti (FAT32).

Failipaigutustabel sisaldab iga kettal oleva faili algusklastrit kannet, mis omakorda sisaldab viita järgmisele failiga seotud klastrile ja nii edasi, kuni faililõpu klastrini.

FAT on tänapäeval kõige levinum failisüsteem. Seda kasutavad nii mäluulgad, mälukaardid kui ka teised väiksema andmemahuga andmekandjad. Seda failisüsteemi



toetavad enamik tänapäeva operatsioonisüsteeme ning toe otsesel puudumisel on võimalus installeerida lisapaketid.

Käesolevas bakalaureusetöös on keskendunud FAT32 failisüsteemile, seega mälumoodulisse sisestatud kaardid peavad olema vormindatud FAT32 failisüsteemi. Muul juhul ei tuvasta mälumoodul mälukaarti ja seega ei ole võimaline teostama failioperatsioone. Failisüsteem on valitud põhjusega, kuna tänapäeval on laialt levinud SD mälukaardid, mis on suurema andmemahuga kui 2 GB. Lisaks, kuna kirjutatavad failid on andmemahult väikesed, siis on oluline väike klasteri suurus.

FAT16 ja FAT32 iseloomustamiseks on toodud alljärgnevas tabelis ära failisüsteemide klasterite erinevused. (Tabel 5) Väikesemahuliste failide puhul tuleb kasuks väiksem klasteri suurus, kuna suurema klasteri korral jääb kasutamata suur osa vabast mälust. See on tingitud sellest, et iga fail kasutab vähemalt ühte klasterit.<sup>[15]</sup>

**Tabel 5. FAT failisüsteemide klasterite suurused.**

Ketta suurus	Vaikimisi FAT16 klaster	Vaikimisi FAT32 klaster
260 MB – 511 MB	8 KB	Pole toetatud
512 MB – 1023 MB	16 KB	4 KB
1024 MB – 2 GB	32 KB	4 KB
2 GB – 8 GB	Pole toetatud	4 KB
8 GB – 16 GB	Pole toetatud	8 KB
16 GB – 32 GB	Pole toetatud	16 KB
> 32 GB	Pole toetatud	32 KB

## 1.8 Ühendusvõimaluse leidmine

Bakalaureusetöö eesmärgiks oli ühendusvõimaluse leidmine NXT juhtploki ning loodava mälumooduli vahel. Ühendusvõimaluse leidmiseks oli vaja uurida NXT poolt pakutavaid võimalusi, mida on kokku kolm.

Esimeseks võimaluseks oleks kasutada juhtploki nelja anduri- ehk sisendporti. Neljast üks, mis kannab nime „Port 4“, on võimaline kasutama kahte liidest: I<sup>2</sup>C ja RS-485. NXT juhtploki I<sup>2</sup>C liidese maksimaalne kiirus 9600 b/s ja RS-485 maksimaalne kiirus umbes 0,9 Mb/s. Mälumooduli disanimisel tuli arvesse võtta selle eesmärk – andmete kogumine. Seega ühe pordi kasutamine võtaks võimaluse sellelt andmeid koguda.

Teiseks võimaluseks oleks kasutada NXT juhtploki USB liidest, mis ei võtaks ära andmete kogumise võimalust portidelt ja oleks esmapilgul kõige parem lahendus, mis

nõuaks vaid USB kaabli ühendamist seadmete vahel. Probleemiks osutub, aga see kuidas on NXT juhtplokis realiseeritud USB liides. Nimelt töötab NXT USB ainult *slave* ehk alama režiimis ja vajab *host* ehk peremehe rolli täitvat seadet. Selleks on vaja muuta NXT ametlikku püsivara, mis muudaks mälumooduli arenduse raskemaks. Hea oleks luua lahendus, mis ei nõua püsivara muutmist ja kasutakaks ametlikku püsivara.

Kolmandaks võimaluseks on kasutada NXT Bluetooth juhtmevaba ühendust, mis ei hõivaks ühtegi NXT porti ja ei võtaks ära ka võimalust ühendada NXT juhtplokki teiste seadmetega. Lisaks on NXT Bluetooth palju paindlikum ühenduste suhtes kui ükski teine ühendamisviis, kuna ta ei vaja kasutamiseks püsivara muutmist ja kasutajale versiooni vahetamist.<sup>[13]</sup>

## **1.9 LEGO MINDSTORMS NXT Bluetooth**

NXT juhtplokk toetab juhtmevaba ühendust, kasutades selleks Bluetooth'i kiipi CSR BlueCore™ 4 versioon 2. NXT juhtplokk suudab juhtmevabalt ühenduda kuni kolme seadmega korraga, aga suhtlust saab läbi viia vaid ühega. Selleks, et vähendada Bluetooth mooduli võimsuse tarbimist, on see rakendatud teise klassi Bluetooth seadmena, mis tähendab, et kommunikatsiooniradius on umbes 10 meetrit.

LEGO MINDSTORMS NXT Bluetooth kasutab SPPd (*Serial Port Profile*), mis on RFCOMM protokoll. See emuleerib jadaliidest ning asendab standardit RS-232. NXT juhtplokis olev Bluetooth kiip suhtleb otse ARM7 protsessoriga, kasutades selleks UART liidest kiirusega 480,6 kb/s.

Bluetooth juhtmevaba ühendust korraldab kiip CSR BlueCore™ 4 v2.0, millel on lisaks EDR (*Enhanced Data Rate*) süsteem. Vastaval kiibil on järgnevad omadused: toetab SPP protokoll, töötab 26 MHz taktsagedusega, sisaldab 47 KiB RAM ning kasutab puhvrina 8 Mb FLASH väälmälu.

NXT Bluetooth andmelehes on kirjeldatud SPP protokollis kaks puudust:

- ARM protsessorilt saab suuremaid andmepakette vastu võtta väikese ajalise erinevusega.
- BlueCore kiip vajab saatja ja vastuvõtja režiimi muutmise ajal umbes 30 ms ajalist vahet.

Bakalaureusetöös tuleb arvestada eespool väljatoodud puudustega. Probleemi lahenduseks on NXT-G programmeerimisploki loomisel viivise sisse toomine. Lisaks oleks sobilik kasutada Bluetooth käsuplokki, mis ei nõua tagastatavat paketti. Sel juhul ei pea BlueCore vahetama saatja ning vastuvõtja režiimi vahel.

NXT kasutab andmete saatmiseks ja vastuvõtmiseks andmepaketti maksimaalse pikkusega 64 baiti. Tavaliselt lisatakse vastuvõetava paketi esimesele kahele baidile informatsioon paketi pikkuse kohta. Järgnevad baidid kuuluvad käsu tüübile, käsule endale ning vastava käsu defineeritud baitidele (Tabel 6).<sup>[12]</sup>

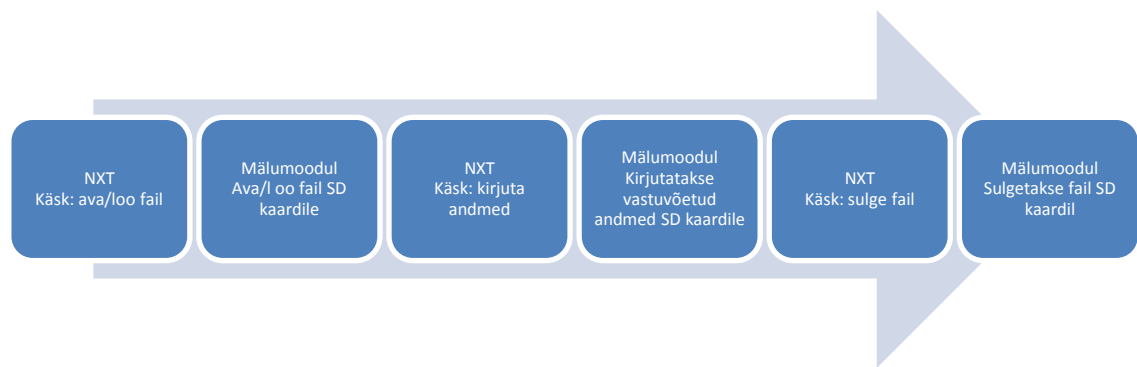
**Tabel 6. LEGO MINDSTORMS NXT andmepakett.**

Bait 0 Pikkus, LSB	Bait 1 Pikkus, MSB	Bait 2 Käsu tüüp	Bait 3 Käsk	...	Bait n
-----------------------	-----------------------	---------------------	----------------	-----	--------

### 1.9.1 Bluetooth ülema-alluva-süsteem

Mälumooduli ja NXT vahel peab eksisteerima *master* ehk ülema ja *slave* ehk alama süsteem. Selleks, et mälumoodulit saaks ühendada NXT juhtplokiga, on vajalik mälumoodulil käituda alamana ehk *slave* seadmena. Ühenduse loob NXT ja käitub seega kui ülem. Peale ühenduse loomist on kaks seadet omavahel ühendatud ja alam hakkab ootama ülemalt käske.

Mälumoodul hakkab käsu saabumisel vastavalt faili avama, kirjutama või sulgema. Protsess on ühesuunaline – NXT juhtplokilt saadetakse käsk ja andmed, millele mälumoodul ei tagasta kättesaamise kinnitust vaid hakkab kohe saadud andmepaketti töötleva. (Joonis 3)



**Joonis 3. NXT ja mälumooduli ühesuunaline suhtlus.**

NXT ja mälumooduli vahelise ühenduse loomiseks kasutakse eraldi Bluetooth moodulit [6], mis ühendatakse mälumoodulile.

### **1.9.2 Mälumooduli Bluetooth moodul**

Valitud Bluetooth moodul koosneb kahest osast. Esimene osa on Bluetooth moodul EGBT-046S, mis on joodetud teise osa, trükkplaadi, külge. Trükkplaadil on ühele küljele välja toodud mooduli viigud ning lisatud valgusdiod, mis aitab tuvastada seadme ühenduvust.

Vaikesätete kohaselt kasutab seade suhtluseks boodikiirust 9600, kaheksat andmebiti ning ühte stoppbitti. Paarsuse ja kätlemise (*handshake*) bitte ei arvestata. Mooduli kommunikatsiooniraadius on 10 meetrit, seega tegemist on klass 2 Bluetooth seadmega.

Bluetooth moodul kasutab toitepingeks 3,3 V ning suhtleb 3,3 V loogikaga. Mooduli ühendamiseks on vaja RXD viik kokku viia mikrokontrolleri TXD viiguga ning TXD viik ühendada mikrokontrolleri RXD viiguga.<sup>[6]</sup>

## 2. Tulemus

### 2.1 Mälumoodul

#### 2.1.1 Elektroonika skeem

Mälumooduli tähtsaimaks komponendiks on mikrokontroller ATmega32U4, mille ülesandeks on vahetada andmeid reaallaja kella, Bluetooth mooduli ja SD kaardi vahel. Mälumooduli elektroonika skeem on välja toodud lisa 2.

Mikrokontroller kasutab välist kristalli taktsagedusega 16 MHz. Kristallile on lisatud kondensaatorid mahtuvusega 18 pF, et vähendada müra ja tagada stabiilsem väljundsignaal.<sup>[3]</sup>

Moodulil on kokku kaks nuppu, mis on ühendatud kontrolleri RESET ja HWB väljaviigu külge. Nupud on lisatud RESET signaali andmiseks või programmeerimiseks vajalikku režiimi, alglaadurisse (*bootloader*), viimiseks. Mõlemad nupud on seatud kõrgesse olekusse, kasutades 10 k $\Omega$  *pull-up* takistiteid ning lahtisidestatud 100 nF kondensaatoritega korduvate lülituste vältimiseks.

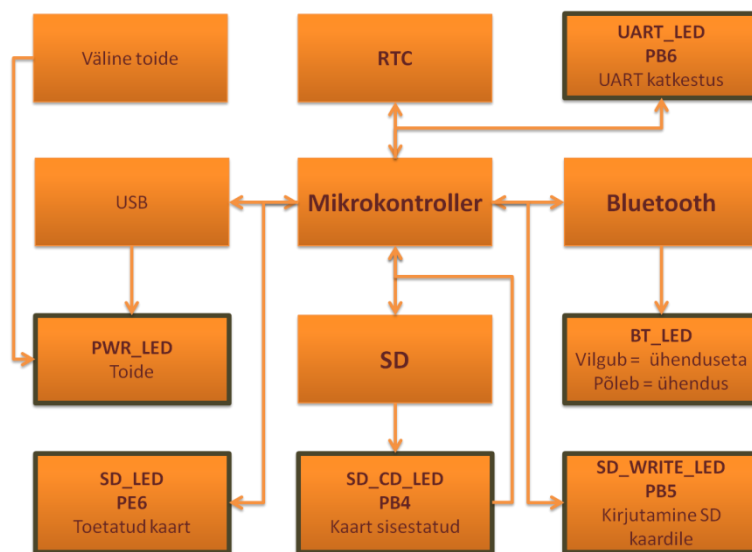
Reaallaja kella ehk RTC (*Real Time Clock*) jaoks kasutatakse kontrolleri TWI liidest, mis sarnaneb I<sup>2</sup>C liidesele<sup>[388]</sup>. Kella juhtimiseks kasutatakse TWI liidest taktsagedusel 100 kHz. TWI kaks liini: SDA ja SCL on seatud kõrgeks, kasutades 3,3 k $\Omega$  *pull-up* takistiteid ja RTC toide on lahtisidestatud 100 nF kondensaatoriga. Lisaks kasutab reaallajakell väärtuste säilitamiseks ja kellatakti tagamiseks tagavara patareid toitepinge väärtusega 3 V ning 32,768 kHz kristalli kellatakti jaoks. Seadme kompaktsuse tagamiseks on kasutatud üherakulist 12 mm läbimõõduga patareid (CR1220).<sup>[14]</sup>

Moodulile on lisatud loogika taseme teisendi (*hex high-to-low level shifter*, edasi „loogika teisendi“). Vastav kiip on vajalik, et teisendada kontrollerist tulev 5 V loogika 3,3 V loogikale SD kaardi ja Bluetooth mooduli jaoks. Kiip kasutab toitepingeks 3,3 V ja vastavalt sellele teisendab sisendsignaale toitepingele vastavalt välissignaale. Toitepinge on lahtisidestatud 100 nF kondensaatoriga. Kasutamata jäänud sisendviigud on ühendatud maaga, kuna tegemist on CMOS seadmega, mida on kergesti võimalik kahjustada staatilise elektriga.<sup>[2]</sup>

Bluetooth moodul kasutab 3,3 V loogikat, seega RXD viik tuleb esmalt viia läbi loogika teisendi ja teisendada 5 V loogika 3,3 V loogikale. TXD viik on viidud otse mikrokontrollerisse, kuna mikrokontroller oskab töödelda 3,3 V loogika tasemega signaale. Mooduli toitepinge on lahtisidestatud 100 nF kondensaatoriga. [6]

SD kaardiga suhtlust korraldatakse SPI liidese abil, taktsagedusega 8 MHz. SD kaardi ühendamiseks plaadile kasutatakse standardsuurusega SD kaardi adapterit. Kuna mikrokontroller kasutab 5 V loogikat, siis on esmalt vaja SD kaardi sisendviigud teisendada 3,3 V loogikale. Seega liinid MOSI, CS ja SCK tuleb viia läbi loogika teisendi. Liin MISO viiakse otse mikrokontrollerisse ja ei vaja teisendamist. SD kaardi toide on lahtisidestatud 100 nF kondensaatoriga. [7]

Moodulile on kokku lisatud 5 valgusdiodi, et tagada kasutajale tagasiside mälumoodulis toimuvate protsesside kohta. Valgusdiodidele on lisatud 1,8 k $\Omega$  takistid, et vältida liigset voolutugevust. Lisana on üks valgusdiodi veel Bluetooth moodulil, mis näitab kas seade on ühendatud või mitte. (Joonis 4)



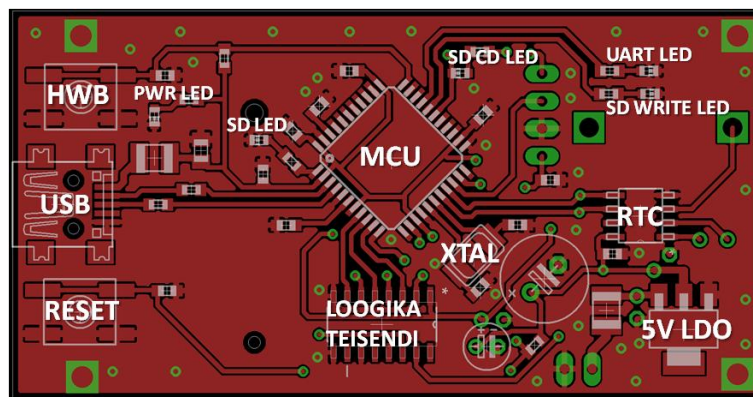
**Joonis 4. Mälumooduli valgusdiodid.**

Moodulil on programmeerimiseks USB port, mida saab kasutada ka seadmele toitepinge andmiseks. USB toiteliinile on lisatud 0,5 A taastuv polümeerkaitse, mis kaitseb süsteemi liigse voolutugevuse ja läbipõlemise eest. Lisaks on toiteliin lahtisidestustatud 10  $\mu$ F kondensaatoriga. USB andmeliinidele on lisatud 22  $\Omega$  terminaatoritakistid, mis ei lase tekkida andmete tagasipeegeldumisel.

Moodulit saab toita kahel viisil: USB kaudu arvutist või välise alalisvooluga toiteallika kaudu, mille toitepinge võib olla vahemikus 6,4 - 16 V. Väline toitepinge peab olema suurem LDO (*Low Dropout*) pingeregulaatori (edasi „LDO“) pingelangust (*drop-out*), mille väärtus on 1,3 V. Moodulil on kokku kaks erineva väljundpingega LDOd: 3,3 V ja 5 V. Vastavad LDOd on kasutusel, et tagada 3,3 V toitepinge Bluetooth moodulile, loogika teisendile ja SD kaardile ning 5 V toitepinge mikrokontrollerile ja reaaliajakellale. LDOle on lisatud 10  $\mu$ F elektrolüütcondensaatorid, mis stabiliseerivad ja vähendavad väljundsignaalis olevat müra.

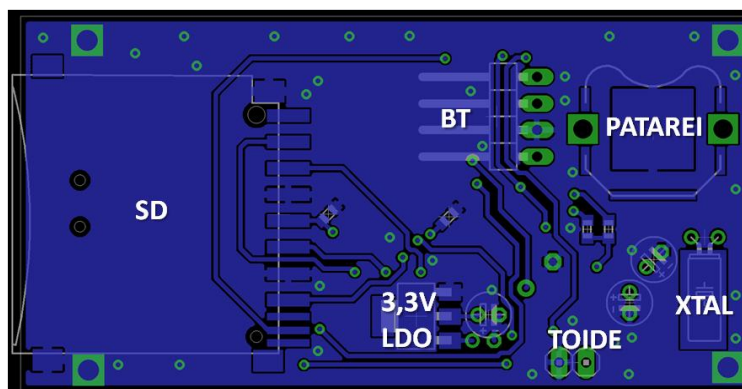
### 2.1.2 Füüsiline ehitus

Mälumoodul on disainitud kahepoolsele trükkplaadile suurusega 69 x 36 mm. Komponentide paigutamisel oli eesmärk seade võimalikult väiksena disainida, arvestades komponentide hulka ja paigutust. Trükkplaadi pealne kiht on toodud joonisel 5 ja alumine kiht joonisel 6. Kasutatud komponentide nimekiri on kirjeldatud tabelis 7.



Joonis 5. Trükkplaadi pealne kiht.

Trükkplaadi pealmisele kihile plaadi keskele on paigutatud mikrokontroller ATmega32U4 (MCU) ning loogika teisendi 74HC4050D (LOOGIKA TEISENDI). Lisaks jääb pealmisele kihile veel reaaliajakell (RTC), USB miniB pesa (USB), 5 V LDO pingeregulaator (5V LDO), kaks nuppu (RESET ja HWB), 16MHz kristall (XTAL) ning valgusdiodid kasutajale informatsiooni tagamiseks (Joonis 4. Mälumooduli valgusdiodid.). (Joonis 5)



**Joonis 6. Trükkplaadi alumine kiht.**

Alumisele kihile on paigaldatud SD standardsuurusega mälukaardi adapter (SD), 3,3 V LDO pingeregulaator (3,3V LDO), reaajakella kristall (XTAL) ja -tagavara patarei hoidja (PATAREI), välise toiteallika konnektor (TOIDE) ning kinnitus (BT) Bluetooth mooduli jaoks. (Joonis 6)

**Tabel 7. Mälumooduli komponentide tabel.**

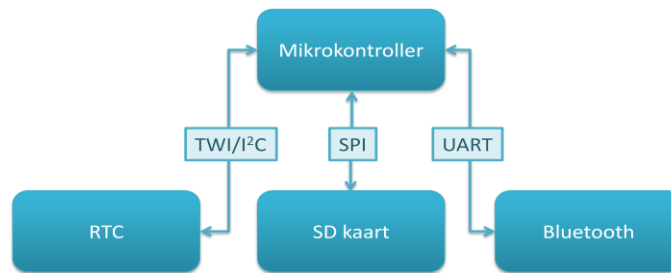
Komponent	Väärtus	Pakk	Kogus
12 mm patarei hoidja	Keystone 3001	Läbiauk	1
3,3 V LDO pingeregulaator	AP1117E33G-13	SOT223	1
5 V LDO pingeregulaator	AP1117E50G-13	SOT223	1
Bluetooth moodul	EGBT-046S		1
Elektrolüüt kondensaator	10 $\mu$ F 16 V	Läbiauk 4 x 5 mm	4
Elektrolüüt kondensaator	100 $\mu$ F 25 V	Läbiauk 5 x 8 mm	1
Kondensaator	10 $\mu$ F	SMD 0805	1
Kondensaator	1 $\mu$ F	SMD 0603	1
Kondensaator kristallile	18pF	SMD 0603	2
Mikrokontrolleri kristall	16 MHz	SMD	1
Reaajakella kristall	32,768 kHz	TC38H	1
Lahtisidestus kondensaator	100 nF	SMD 0603	11
Valgusdiodi takisti	1,8 k $\Omega$	SMD 0603	5
Loogika teisendi	74HC4050D	SOIC16	1
Mikrokontroller	ATmega32U4	44TQFP	1
Nupp	SMT	EVQQ2 – 6 mm	2
Nurk-pesapiikriba	1x04/90	2,54 mm	1
Piikriba	1 x 02	2,54 mm	1
Nupu <i>pull-up</i> takisti	10 k $\Omega$	SMD 0603	2
TWI <i>pull-up</i> takisti	3,3 k $\Omega$	SMD 0603	2
Punane valgusdiod	LED 20 mA	SMD 0603	5
Reaajakell	DS1307z+	SOIC8	1
SD adapter	Te Connectivity SD	SMD	1
Taastuv polümeerkaitse akule	1 A PTC	SMD 1210	1
Taastuv polümeerkaitse USB'le	0,5 A PTC	SMD 1210	1
USB andmeliini takisti	22 $\Omega$	SMD 0603	2
USB-mini B konnektor	USB-MINIB	SMD	1



### 2.1.3 Programmikood

Mälumooduli programmikood on kirjutatud C programmeerimiskeeles. Kasutatud on CC Dharmani lähtekoodi <sup>[19]</sup>, mida on vastavalt vajadusele muudetud. Lähtekood oli kirjutatud ATmega32 mikrokontrollerile, mis kasutas arvutiga suhtlemiseks RS232 protokoll.

Joonis 7 kujutab mooduli üldist tööpõhimõtet ja kasutatavaid liideseid.



**Joonis 7. Mälumooduli üldine tööpõhimõte**

Koodis on tehtud järgnevad muudatused: Bluetooth mooduli tugi, NXT failinime lugemine, faili kirjutamine koos ajatempliga, faili kirjutamine ilma ajatemplita, UART katkestuste kasutamine ja mittevajaliku koodi eemaldamine.

Programmi kirjutamiseks ja arendamiseks kasutati arenduskeskkonda AVR Studio 5.1. Mikrokontrollerile kirjutamiseks kasutati Atmeli ametlikku tarkvara Flip 3.4.7. Kompileeritud koodi suurus on umbes 20 KB ning see kasutab umbes 0,6 KB ATmega32U4 RAM. Kontrolleri programmeerimiseks kasutati USB liidest.

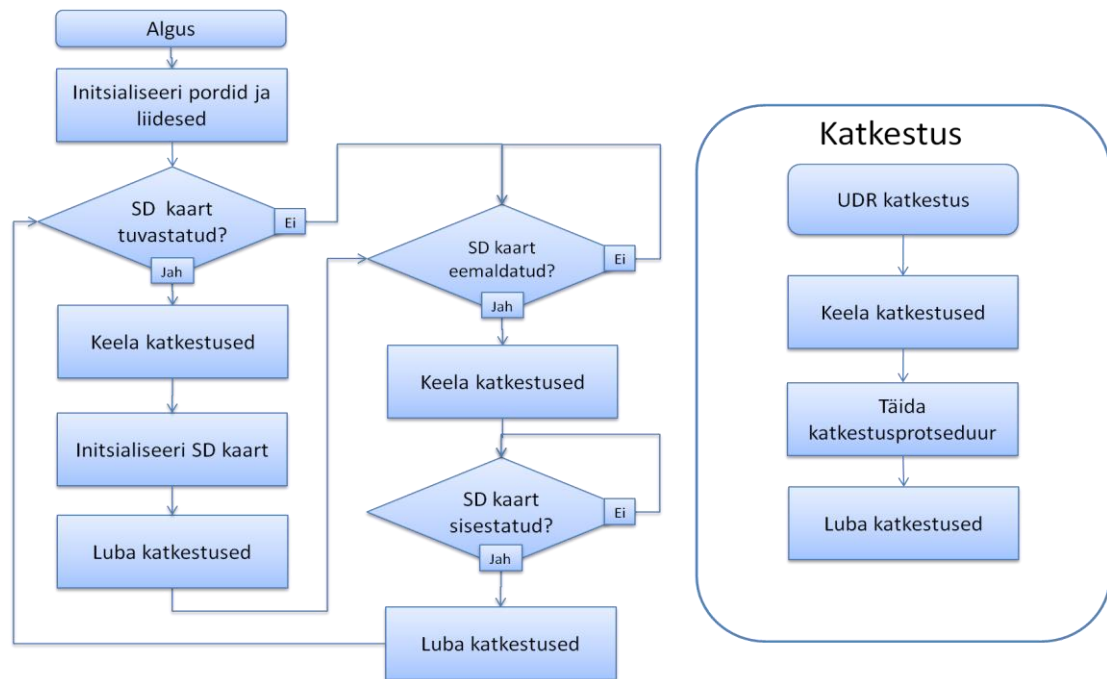
ATmega32U4 programmeerimiseks on vaja see esmalt viia alglaadurisse. Selleks kasutatakse RESET ja HWB nuppe. Peal tuleb hoida RESET nuppu ja siis vajutada HWB nuppu, seejärel tuleb nupud lahti lasta. Kontroller loeb HWB signaali tõusvat fronti ja vastavalt sellele minnakse alglaadurisse.

Programmikood on lisatud tööga kaasas oleval CD plaadil. (Lisa 1)

#### 2.1.3.1 Peaprogramm

Peaprogrammis initialiseeritakse liidesed, pordid ning muutujad. Programmi üheks eesmärgiks on täita katkestusi, mida tekitab Bluetooth moodulilt tulevad signaalid. Teiseks eesmärgiks on SD kaardi tuvastamine ja selle initialiseerimine. Peaprogrammi tööd iseloomustab järgnev plokkiagramm (Joonis 8). Katkestuste rutiin täidetakse

jooksvalt. Kaardi korrapäraseks initsialiseerimiseks keelatakse selle vältel katkestused. Sama moodi toimitakse ka kaardi puudumisel.



**Joonis 8. Peaprogrammi plokkiagramm.**

### 2.1.3.2 SD kaart

SD mälukaardi korrapäraseks tööks on vaja korralda kaardi initsialiseerimine kaardi eemaldamisel ja uuesti sisestamisel. Kaardi initsialiseerimisprotsessi juhitakse peaprogrammis.

Kaardi initsialiseerimisel tuleb kasutada vastavalt SD andmelehele aeglast SPI režiimi taktsagedusega 400 kHz. Peale esimeste käskude saatmist võib SPI taktsageduse tõsta maksimaalse lubatud piirini, mis mälumoodulil on pool mikrokontrolleri taktsagedusest ehk 8 MHz.

Kaardi korrapäraseks initsialiseerimiseks on vajalik FAT32 failisüsteemi vormindatud SD mälukaart. Eduka initsialiseerimise järel süttib ennisti vilkunud või kustunud SD LED (Joonis 5), mis tähendab, et kaart on vormindatud FAT32 failisüsteemi ja on nüüd valmis kasutamiseks. SD kaardi ebaõnnestunud initsialiseerimisel kustub SD LED, ning kaardi eemaldamisel hakkab vastav valgusdiod taas vilkuma. Lisatud on veel SD CD LED (Joonis 5), mis süttib kaardi adapterisse sisestamisel ja annab seega mikrokontrollerile märku, et kaart on ühendatud.

Peale kaardi ettevalmistust hakatakse ootama UART katkestust, mida tekitab Bluetooth moodulilt tulev signaal. Signaali olemasolul minnakse katkestusrutiini ning täidetakse vastavalt käsule kas faili avamine, kirjutamine või sulgemine.

Kaardi ebaõnnestunud initsialiseerimisel võib proovida kaarti välja tõmmata ning uuesti sisestada. See on vajalik kui kaarti ei eemaldata süsteemist turvaliselt, mistõttu kaart võib vajada lähtestamist. Lisaks võib olla ka probleemiks FAT32 failisüsteemi puudumine kaardilt. Seega enne kaardi sisestamist tuleks veenduda, et kaart on vormindatud FAT32 failisüsteemi.

### 2.1.3.3 RTC

Moodul kasutab ajatempli loomiseks reaaliajakella ehk RTCd. Suhtlus mooduli ja RTC vahel toimub, kasutades TWI/I<sup>2</sup>C liidest. Reaaliajakellalt tulev näit salvestatakse kas faili või lisatakse faili loomisel/kirjutamisel FAT kirjesse.

Kasutusel on Maxim DS1307 reaaliajakell, mis tagastab kasutajale sekundid, minutid, tunnid, aasta, kuu ning päeva. Selleks kasutatakse RTC-s asuvaid registreid, mis asuvad aadressidel 00h – 06h. Vastavad registrid on kirjeldatud alljärgnevas tabelis (Tabel 8).<sup>[14]</sup>

```
//NIL
time[8] = 0x00;
//Seconds(1's)
time[7] = (SECONDS & 0x0F) | 0x30;
//Seconds(10's)
time[6] = (SECONDS & 0x70) >> 4 | 0x30;
time[5] = ':';
//Minutes(1's)
time[4] = (MINUTES & 0x0F) | 0x30;
//Minutes(10's)
time[3] = ((MINUTES & 0x70) >> 4) | 0x30;
time[2] = ':';
//Hours(1's)
time[1] = (HOURS & 0x0F) | 0x30;
//Hours(10's)
time[0] = ((HOURS & 0x30) >> 4) | 0x30;
```

**Joonis 9. RTC programmilõik.**

Registrite sisu loetakse eraldi massiivi. RTC-lt saadud andmed teisendatakse massiivi kirjutamisel. Iga loetud registriväärtus teisendatakse vastavalt alltoodud tabelis kirjeldatud kujule (Tabel 8). Näiteks loetakse sekundi kümnendosa, sooritades loogiline NING tehe kuueteistkümnendsüsteemi arvuga 0x0F ning nihutatakse saadud tulemust nelja võrra paremale. Seejärel sooritatakse loogiline VÕI operatsioon kuueteistkümnendsüsteemi arvuga 0x30. Kellaaja teisendust kirjeldab vasakul asuv kood. (Joonis 9)

**Tabel 8. DS1307 registrid.**

Address	Bitt 7	Bitt 6	Bitt 5	Bitt 4	Bitt 3	Bitt 2	Bitt 1	Bitt 0	Funktsioon	Vahe
00h	CH	Sekund			Sekund				Sekundid	00-59
01h	0	12	Tund	Tund	Tund				Tunnid	1-12 + AM/ PM 00 - 23
		24	PM/ AM							
03h	0	0	0	0	0	Päev			Päev	01-07
04h	0	0	Kuupäev		Kuupäev				Kuupäev	01-31
05h	0	0	0	Kuu	Kuu				Kuu	1-12
06h	Aasta				Aasta				Aasta	00-99

#### 2.1.3.4 Bluetooth

Mälumooduli suhtlust Bluetooth mooduliga korraldatakse UART liidesega. Suhtluse loomiseks kasutatakse 9600 boodikiirust, kaheksat andmebitti ning ühte stopp bitti. Paarsuse ja kätlemise (*handshake*) bitte ei arvestata.

Moodulite vaheline andmeedastuskiirus on Bluetooth mooduli vaikekiirus. Sissetulevad andmed kirjutatakse Bluetooth mooduli puhvrissi, mis edastatakse mikrokontrollerile.

## 2.2 Programmeerimisplukkide loomine NXT-G'le

Programmeerimisplukkide loomiseks programmeerimiskeelele NXT-G on vajalik need luua LabVIEW arenduskeskkonnas. Tähtis on kasutada LabVIEW versiooni 7.1, sest vastavas keskkonnas on loodud ka arenduskeskkond NXT-G. LabVIEW'le on lisaks vaja allalaadida ja paigaldada NXT Toolkit, mis sisaldab näiteid ja elemente NXT programmipluki loomiseks.

Plokkide loomiseks kasutati dokumenti „*Creating LEGO MINDSTORMS NXT Software Blocks*“ ja Kait Elbe bakalaureusetöös kirjeldatud pluki loomise protsessi. [9,<sup>11</sup>]

### 2.2.1 BTMemory plokk

Ploki arendamise käigus loodi mitu erinevat versiooni programmeerimisplukke. Ideeks oli kasutada ära kõiki käskke, mida saadetakse mälumoodulile, ja muuta need võimalikult lihtsasti kasutatavaks. Lõpliku lahendusena koguti kogu funktsionaalsus ühte programmeerimisplukki. Lõplik versioon sai nimeks *BTMemory Block*.

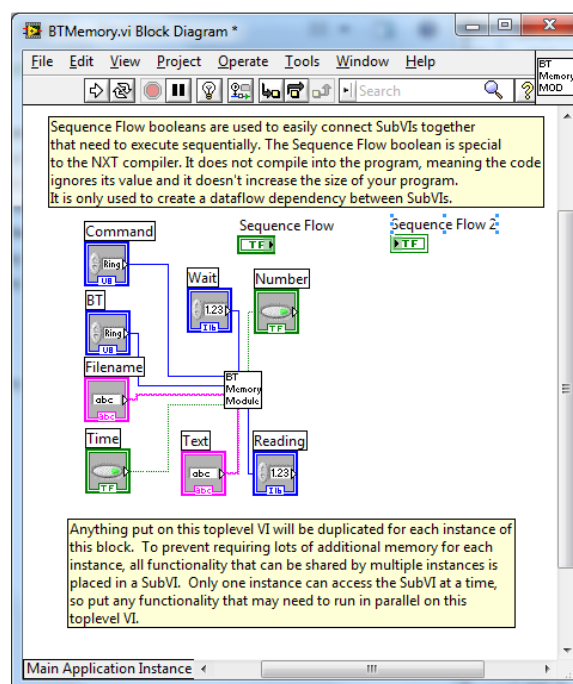
BTMemory plukki on kogutud faili avamine, kirjutamine ning sulgemine. Juurde on lisatud funktsionaalsus ajatempli lisamiseks andmete kogumisel, teksti kirjutamiseks

faili ning Bluetooth ühenduse valik. Ploki toomine NXT-G tarkvarasse on kirjeldatud lisas 3 ning ploki kasutusjuhend koos näiteülesandega lisas 4.

### 2.2.2 ÜlemVI

ÜlemVI on peaprogramm, mida täidetakse. Sellesse paigutatakse kasutatavad alamVID ning teised alamprogrammid.

ÜlemVI (*BTMemory.vi*) kasutab ühte alamVID (*BTMemory Sub.vi*), kuhu on kirjeldatud kogu funktsionaalsus. Kokku on programmil 8 sisendit, mille abil saab muuta ploki funktsionaalsust. ÜlemVI on kujutatud järgneval joonisel. (Joonis 10)

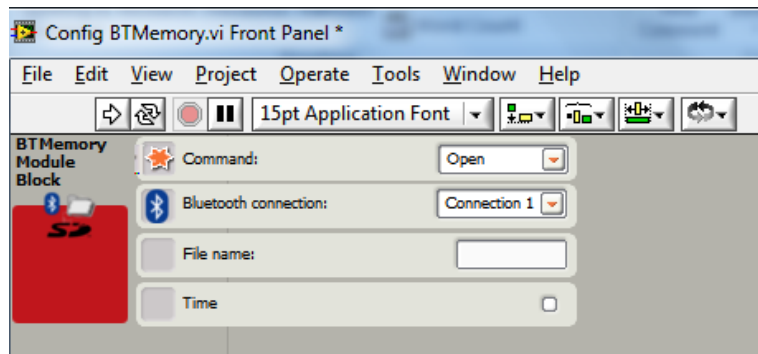


Joonis 10. ÜlemVI - BTMemory.vi

### 2.2.3 Konfiguratsiooni VI

Konfiguratsiooni VIs kirjeldatakse ja luuakse kasutajaliides, mida hiljem kuvatakse MINDSTORMS tarkvaras sätete aknas. Lisaks toimub selles VIs vaikeväärtuste uuendamine.

Konfiguratsiooni VI loomisel tuleb kindlasti disainitav kasutajaliides paigutada ülesse vasakusse äärde. (Joonis 11) Vastasel juhul kuvab NXT-G sätete akna kas nihkes või üldse mitte.

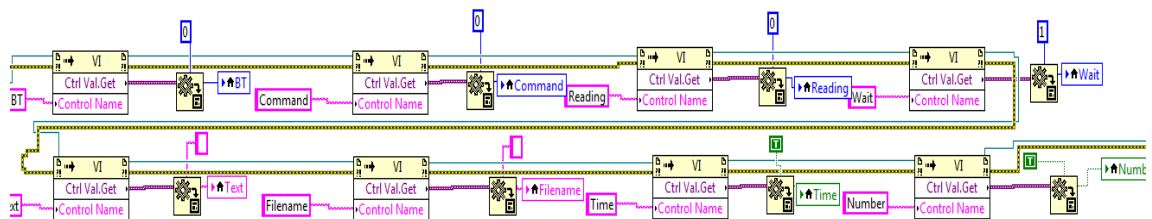


**Joonis 11. Kasutajaliides - Config BTMemory.vi**

Vaikeväärtuste puhul on tähtis kasutada samasid muutujanimesid, mis paiknevad ülemVIs, sest muidu ei uuendata väärtusi.

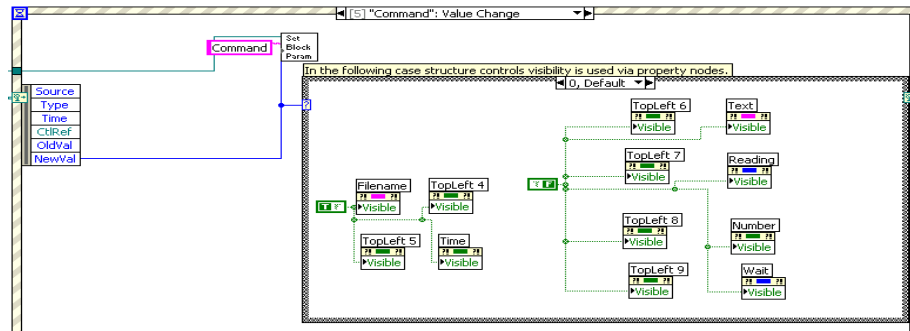
Ploki arenduse käigus oli probleeme kõige rohkem konfiguratsiooni VIs, sest tihtipeale nihkus kasutajaliides, mis tähendas uut salvestamist ja paigutamist. Probleemideta ei läinud ka vaikeväärtuste uuendamine, mis nõudis väga täpset muutujatenimede süsteemi ja kohati ka täpseid andmetüüpe.

Alljärgneval joonisel on toodud vaikeväärtuste seadistamine. Joonisel on kuvatud osa programmist, mis kasutab kasutajaliidese elementide lokaalseid muutujaid ja omistatakse nendele ülemVIs väärtused. Sellega ühendatakse kasutajaliides ja programmi muutujad. (Joonis 12)



**Joonis 12. Muutujate defineerimine - Config BTMemory.vi**

Kasutajaliidesele on lisatud ka andmeväljade muutumine ehk vastavalt käsule kuvatakse kasutajale vaid vajalikud andmeväljad. Kuna vastavale toimingule puudub dokumentatsioon, siis oli vaja see luua. Ülesande lahendamiseks kasutati „Property node“ plokkide, mis loodi iga muutuja jaoks. Sellega muudeti vastavalt käsule andmevälja nähtavust. Loodud lahendus on kirjeldatud järgneval joonisel (Joonis 13)



Joonis 13. Andmeväljade nähtavus - Config BTMemory.vi

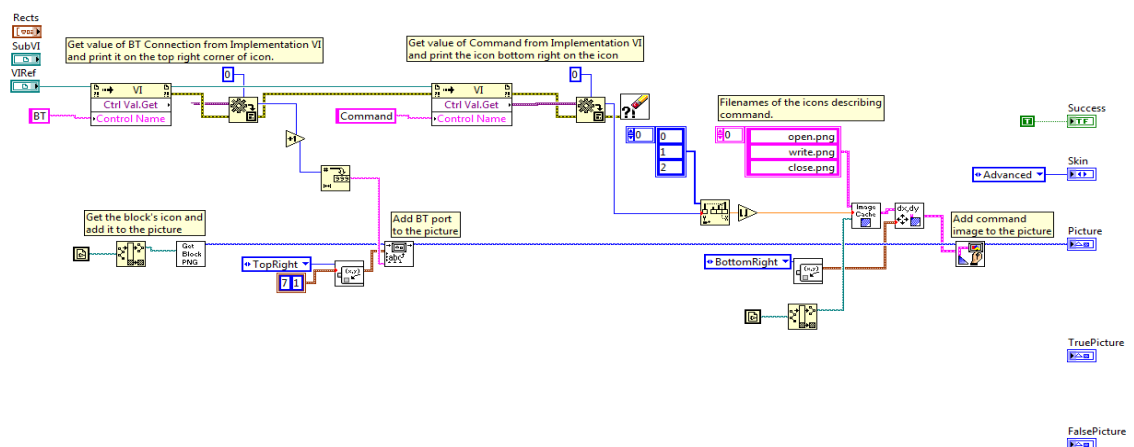
## 2.2.4 Joonistamise VI

Joonistamise VIs kirjeldatakse plokki välimus MINDSTORMS tarkvara plokki diagrammis. Sätestatakse ära ikooni paigutus ja lisafunktsioonid, näiteks reaajas muutuvad ikoonid.

Vahetuva ikooni loomiseks kasutati NXT Toolkitiga kaasasolevat „Full Sensor Block“ näidet, kus oli kirjeldatud anduri valgustugevuse ikooni muutust kui seda liigutada liugribal.

Loodi plokk, mis peale igat käsku (avamine, kirjutamine, sulgemine) muutis plokki välimust. Lisati ka muutuv Bluetooth ühenduse pordi number.

Loodud VI ei kasuta *TruePicture* ja *FalsePicture* väärtusi, kuna need defineeritakse vaid anduriplokkidele. Muutuvad väärtused *BT* ning *Command* kutsuvad esile plokki ikooni muutuse, vahetades vastavalt numbrit ja käsu ikooni. (Joonis 14)



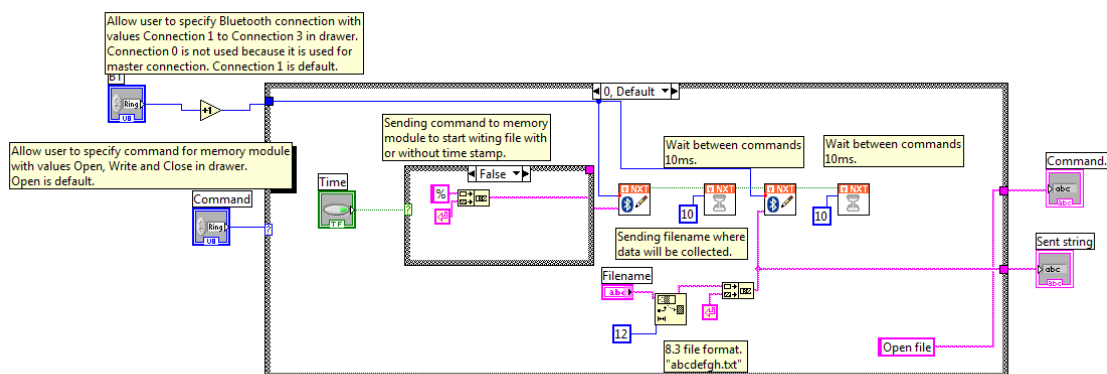
Joonis 14. Joonistamise VI - Draw BTMemory.vi

### 2.2.5 AlamVI

AlamVI kasutatakse programmi loogika kirjeldamiseks. Loodud plokis kasutatakse ühte alamVI, milles kirjeldatakse mälumoodulile vajalike andmete saatmist.

Kasutatud on „case“ struktuuri, mis vahetab vastavalt käsule kirjeldatud loogikat. Lisatud on joonis faili avamise kohta, kus on kirjeldatud aja lisamist faili, failinime saatmist seadmele ja käsu saatmist seadmele faili avamiseks. (Joonis 15)

Iga Bluetooth käsu vahele on pandud viivis, mis on vajalik vastavalt NXT Bluetooth andmelehele.<sup>[12]</sup> Töö protsessis testiti ka ilma viiviseta koodi, kuid see ei töötanud korrapäraselt.



Joonis 15. Alam VI faili loomine ajatemplita - BTMemory Sub.vi

## 2.3 Mälumooduli tarkvara silumine, testimine ning tulemused

Mälumooduli silumine ja testimine olid käesoleva bakalaureusetöö tähtis osa. LEGO MINDSTORMS NXT ja mälumooduli suhtluse tagamiseks oli vaja esmalt kontrollida mälumooduli korrapärasest tööd. Selleks testiti mälumooduli suhtlust algselt arvutiga, millel oli olemas Bluetooth tugi. Testimiseks kasutati mälumooduli USB porti, terminal tarkvara ning selle Bluetooth moodulit.

### 2.3.1 Testimine arvuti ja terminal tarkvaraga

Virtuaalse jadaliidese loomiseks kasutati PJRC Teensy USB Serial teeki<sup>[17]</sup>, mis lisati programmikoodile, et selgitada saadetud ja vastuvõetud andmete täpsust. Teegi abil tekitati üle USB arvutisse virtuaalne jadaliides (COM port), mida oli võimalik kuulata ja



lugada terminal tarkvara abil. Terminal tarkvarana kasutati Termite või Tera Term programme.

Mälumoodul ühendati esmalt Bluetooth kaudu arvuti Bluetooth mooduliga. Käivitati terminal tarkvara ning avati loodud COM port. Järgnevalt avati USB läbi tekitatud COM port, mida hakati kuulama. Järgnevalt saadeti arvuti Bluetooth kaudu sõnumeid mälumoodulile, millele vastavalt pidi USB COM pordi terminali aknas tekkima saadetud sõnum. Nõnda kontrolliti sissetulevate ja väljuvate sõnumeid. Lisaks testiti koodis olevaid funktsioone, nagu faili kirjutamine, reaajakella aja tagastamine ning uuriti tekkinud tagasisidet virtuaalse jadaliidese terminali väljundi aknast.

### **2.3.2 Testimine NXT juhtplokiga**

Peale arvutiga testimist oli vajalik testida mälumoodulit. Selleks loodi esmalt NXT juhtplokile programmikood, kasutades loodud plokki ning NXT-G tarkvara. Seejärel ühendati juhtplokk mälumooduliga.

Testimise käigus sooritati järgnevad katsed:

- loodi mitmeid erineva nimega faile
- kirjutati mitu korda sama nimega faili
- eemaldati SD mälukaart mälumoodulist ning sisestati see uuesti
- testiti kaardi initsialiseerimist
- analüüsiti loodud faile, leiti nõrgad kohad ja siluti programmi.

Testide tulemusena selgus, et andmete kirjutamisel tuleb eemaldada ASCII koodis olevad esimesed ning viimased sümbolid, mis tulenesid NXT Bluetooth andmepaketi ülesehitusest ja käskudest. NXT juhtplokki andmete saatmise käsukood, mis ei nõua tagastust, on kuueteistkümnendsüsteemis 0x80. Tekstifailis kuvatakse see sümbolina €.

### **2.3.3 Bluetooth ühenduse testimine**

Bakalaureusetöös kasutatakse suhtlemiseks Bluetooth ühendust. Tegemist on juhtmevaba ühendusega, seega sellel on leviala piirang. Mõlemad Bluetooth seadmed kuuluvad teise klassi, mis tähendab, et seadmete vaheline maksimaalne kaugus ulatub 10 meetrini. Ühenduse kontrollimiseks tehti katseid, mis koosnesid andmete saatmisest erinevate vahemaade tagant.

Testimiseks jäeti mälumoodul statsionaarseks seadmeks kui NXT juhtplokiga liiguti seadmest järk-järgult kaugemale. Koostati programm, mis saatis mälumoodulile 200 näitu sekundiliste vahedega.

Ühenduse olemasolul õnnestus saata kõik andmed. Mõnel üksikul korral toimus faili kirjutamisel moonutus. Mõõtmistulemustest selgus, et klass 2 defineeritud seadmete kaugus on vastavuses kuid siiski ei ole soovitatav kasutada seadet kaugemal kui 7 meetrit.

Suhtlust testiti ka liikuvast keskkonnas ning testidest selgus, et seadmed suudavad andmeid vahetada. Siiski tekkis kohati segaseid tulemeid, kuna mõni tähemärk jäi saatmata.

#### **2.3.4 Mälumooduli voolutarbimine**

Mälumooduli toitepingeks saab kasutada arvuti USB toitepinget kui ka välist toidet, mille toitepinge peab jääma vahemikku 6,4 V – 16 V. Mooduli voolutarbimist saab mõõta kui ühendada jadamisi vooluallika ja mälumooduli vahele ampermeeter.

Testimiseks kasutati multimeetrit Fluke 117, mis seadistati mõõtma voolutugevust. Ampermeetrina töötav multimeeter ühendati toiteallika ja mälumooduli vahele jadamisi ning alustati andmete kogumist. Andmeid koguti 10 minuti jooksul, sooritades erinevaid teste.

Andmete kogumise käigus selgitati välja järgnevad tulemused:

- Seade tarbib keskmiselt 50 mA kui Bluetooth ühendust ei ole teise seadmega loodud.
- Seadme voolutarbimist mõjutab enim Bluetooth moodul, mis leviedastuse (*broadcast*) režiimis tarbib kõige rohkem voolu.
- Maksimaalne mõõdetud voolutugevus on 75 mA, mille tarbimine on mõõdetud momendil kui sisestatakse SD kaart ning Bluetooth moodul pole ühendatud
- Seade tarbib keskmiselt 32 mA juhul kui see on ühendatud teise Bluetooth seadmega ning SD kaart on initsialiseeritud ja kirjutamist ei toimu.
- Seade tarbib keskmiselt 45 mA juhul kui see on ühendatud teise Bluetooth seadmega ning SD kaart on initsialiseeritud ja toimub andmete salvestamine.

Vastavalt andmelehtedele peaks Bluetooth moodul tarbima maksimaalselt 40 mA<sup>[6]</sup>, SD kaart 100 mA<sup>[18]</sup> ning mikrokontroller umbes 15 mA<sup>[3]</sup>.

Mälumooduli keskmiseks voolutarbeks on 43 mA, mis tähendab, et kui seadet toita 6LR61 tüüpi 9 V leelispatareiga (*alkaline*), mille tüüpiline mahutavus on 565 mAh, suudab seade töötada järjepidevalt umbes 9 – 10 tundi.

$$Patarei keskmine eluiga = \frac{Patarei mahutavus [mAh] * 0,7}{Patarei keskmine voolutarve [mAh]}$$

### **2.3.5 Testimine erinevate kaartidega**

Testimiseks valiti kokku 4 erinevat SD kaarti. Kõikide kaartidega sooritati järgnevaid teste: kirjutamine, lugemine, failide kuvamine, mälu mahu lugemine, initsialiseerimine, kustumine. Testiti ka NXT juhtplokilt kirjutamist.

Testitud mälukaardid: 2 GB microSD, Toshiba 4 GB SDHC klass 4, 8 GB microSDHC klass 4 ning Kingston 16 GB microSDHC klass 4.

Testide tulemustest selgus, et kõik kaardid olid mälumooduli poolt toetatud. Sooritatud testid õnnestusid kõikide kaartidega.

## Kokkuvõte

Käesoleva bakalaureusetöö eesmärgiks oli leida viis, kuidas suurendada robotikaplatvormi LEGO MINDSTORMS NXT piiratud mälu kogust ning dokumenteerida töö protsessi nii, et seda oleks võimalik kasutada edasiste moodulite loomiseks.

Töö esimeses osas tutvustati lähemalt LEGO MINDSTORMS NXT platvormi ning tänapäeval laialdast kasutust leidvat SD mälukaarti. Analüüsiti NXT ühendamisvõimalusi ning valiti välja parim lahendus, mille alusel hakati disainima lisamälu moodulit.

Ühendusviisi leidmiseks analüüsiti NXT juhtplokki poolt toetatud protokolle ja leiti nende hulgast sobivaim lahendus. Ühendusviisiks valiti Bluetooth. Lisamälu andmekandjaks valiti SD mälukaart, kuna tegemist on väga laialt levinud ja hästi dokumenteeritud andmekandjaga.

Töö teises osas kirjeldati mälumooduli arendusprotsessi. Loodud mälumoodul kasutab andmekandjana SD mälukaarti ning NXT juhtplokiga suhtlemiseks Bluetooth ühendust. Mälumoodulile lisati reaalkell, et lisada kogutavatele andmele juurde ajatempel. Lisaks loodi NXT-G tarkvarale lisamälu kasutamiseks programmeerimisplakk. Kirjeldati programmeerimisplaki loomise protsessi ning kirjutati sellele õppematerjalina kasutusjuhend. Teise osa lõpus kirjeldati loodud mälumooduli testimistulemusi ja –protsesse.

Bakalaureusetööna loodud failid ning materjalid on lisatud kaasasolevale CD plaadile (Lisa 1)

# **Creating an external data drive for MINDSTORMS robotics platform**

Bachelor's Theses

Kait Krull

## **Summary**

LEGO MINDSTORMS is a robotics platform which was developed to help students understand science and to get them more involved. It is a versatile platform which lets people to get more out of LEGO by using it to learn the essence of science and technology.

The main goal of the present bachelor's thesis was to find a way to increase the limited memory of a LEGO MINDSTORMS NXT robotics platform and document this process to be used for development of new extension modules.

As a result of this thesis an external data drive - a memory module - was created for the LEGO MINDSTORMS NXT robotics platform. The thesis was divided into two major parts.

First part of the thesis described LEGO MINDSTORMS platform, SD memory card and how a connectivity option was chosen and why. As a result Bluetooth was chosen to make a connection between the memory module and SD card as a data drive.

Second part of the thesis describes how the device was developed, how the NXT-G programming blocks were created, what was involved in the process and what the results of the tested device were.

Files and materials created as a result of this bachelor's theses are included with the CD (Lisa1).

## Kasutatud allikad

1. „LEGO MINDSTORMS NXT Features and Limitations“, [http://cs.uwindsor.ca/~malovicd/499football/features\\_limitations.pdf](http://cs.uwindsor.ca/~malovicd/499football/features_limitations.pdf) 16.11.2012.
2. 74HC4050 andmeleht, <http://www.farnell.com/datasheets/1701315.pdf> 05.05.2013.
3. Atmel, ATmega32U4 andmeleht, <http://www.atmel.com/Images/doc7766.pdf> 05.05.2013.
4. D. Alimisis, "Robotics in Education & Education in Robotics: Shifting Focus from Technology to Pedagogy," in *Proceedings of the 3rd international conference on Robotics in Education*, RiE2012. Sept. 2012., pp. 7-14. MatfyzPress, Czech Republic.
5. D. Mindell, "LEGO MINDSTORMS: The Structure of an Engineering (R)evolution," Dec. 2000, pp. 3-5. <http://web.mit.edu/6.933/www/Fall2000/LEGOMINDSTORMS.pdf> 16.11.2012.
6. EGBT-046S andmeleht, <http://www.rasmicro.com/Bluetooth/EGBT-045MS-046S%20Bluetooth%20Module%20Manual%20rev%201r0.pdf> 05.05.2013.
7. F. Foust, „Secure Digital Card Interface for the MSP430“, Michigan State University, Dept. Of Electrical and Computer Engineering, 2004. [http://alumni.cs.ucr.edu/~amitra/sdcard/Additional/sdcard\\_appnote\\_foust.pdf](http://alumni.cs.ucr.edu/~amitra/sdcard/Additional/sdcard_appnote_foust.pdf) 05.05.2013
8. I2C Bus - TWI Bus, <http://www.i2c-bus.org/twi-bus/> 12.03.2013.
9. K. Elbe, „LEGO MINDSTORMS NXT platvormi GPS laiendusmoodul“, Tartu 2010, lk 20 – 22, [http://www.ims.ut.ee/~karl/thesis2011/bak/kait\\_elbe\\_bak\\_2011.pdf](http://www.ims.ut.ee/~karl/thesis2011/bak/kait_elbe_bak_2011.pdf) 05.05.2013.
10. Kooliroboti projekt, <http://www.robootika.ee/LEGO/projekt/index.php/projektist/> 16.11.2012.

11. LEGO, „Creating LEGO MINDSTORMS NXT Software Blocks“,  
[ftp://ftp.ni.com/evaluation/mindstorms/NXT\\_Creating\\_MINDSTORMS\\_Blocks.pdf](ftp://ftp.ni.com/evaluation/mindstorms/NXT_Creating_MINDSTORMS_Blocks.pdf)  
16.11.2012.
12. LEGO, „LEGO. LEGO MINDSTORMS NXT Bluetooth Developer Kit“; Version  
1.00. ,  
<http://MINDSTORMS.LEGO.com/en-us/Support/Files/Advanced.aspx#Advanced>  
16.11.2012.
13. LEGO, „LEGO. LEGO MINDSTORMS NXT Hardware Developer Kit“; Version  
1.00. ,  
<http://MINDSTORMS.LEGO.com/en-us/Support/Files/Advanced.aspx#Advanced>  
16.11.2012.
14. Maxim, DS1307 andmeleht, <http://www.farnell.com/datasheets/46359.pdf>  
05.05.2013.
15. Microsoft Technet, „FAT16 vs. FAT32“, <http://technet.microsoft.com/en-us/library/cc940351.aspx> 10.02.2013.
16. MIT Meedia Labor. <http://www.media.mit.edu/sponsorship/getting-value/collaborations/MINDSTORMS> 16.11.2012.
17. PJRC Teensy USB, [http://www.pjrc.com/teensy/usb\\_serial.html](http://www.pjrc.com/teensy/usb_serial.html) 10.02.2013.
18. SD kaardi andmeleht,  
[https://www.sdcard.org/downloads/pls/simplified\\_specs/part1\\_410.pdf](https://www.sdcard.org/downloads/pls/simplified_specs/part1_410.pdf) 10.02.2013.
19. SD/SDHC Card Interfacing with ATmega8 / 32 (FAT32 implementation).  
<http://www.dharmanitech.com/2009/01/sd-card-interfacing-with-atmega8-fat32.html> 20.02.2013.
20. What is leJOS NXJ,  
<http://lejos.sourceforge.net/nxt/nxj/tutorial/Preliminaries/Intro.htm> 12.03.2013.

## **LISAD**

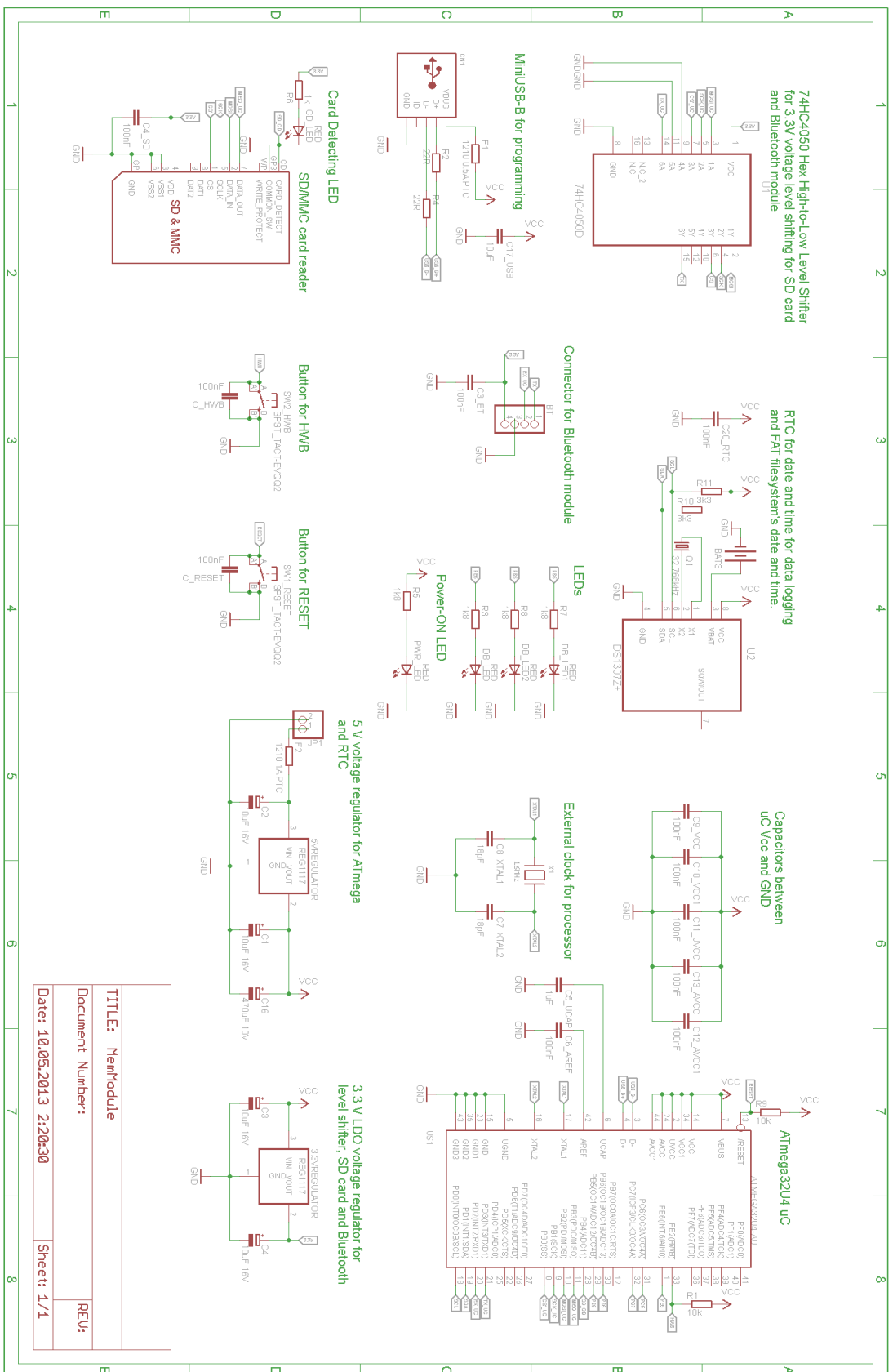
### **LISA 1 – CD**

CD sisu:

1. Mälumooduli elektroonika skeem ning trükkplaadi disain – kataloogis *hardware*
2. Mälumooduli programmikood ja projekt – kataloogis *src/SD\_BT\_RTC*
3. LEGO MINDSTORMS NXT ploki programmikood – kataloogis *src/BTMemory*
4. Andmelehed – kataloogis *datasheets*
5. Käesolev bakalaureusetöö – *Baka\_Kait\_Krull.pdf*



## Lisa 2 – Elektroonika skeem

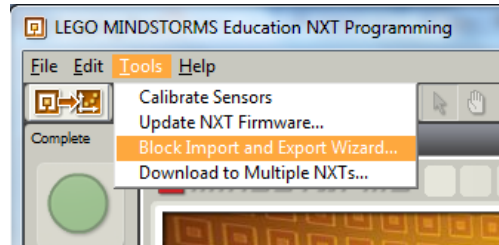


TITLE: Memmodule	
Document Number:	REV:
Date: 10.05.2013 2:20:30	Sheet: 1/1

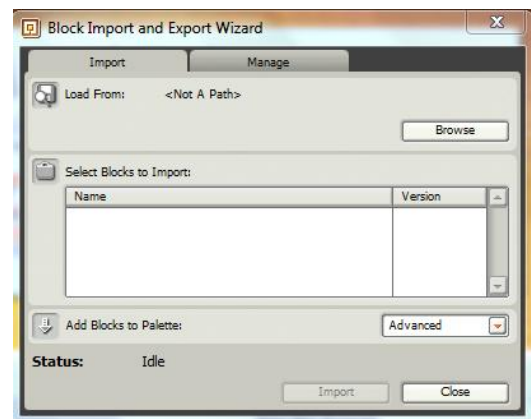
### LISA 3 – BTMemory plokki paigaldamine NXT-G 2.0 tarkvarasse.

Ploki edukaks paigaldamiseks NXT-G tarkvarasse on vaja täita järgmised etapid:

1. Avada programm **NXT-G 2.0 Programming**.
2. Valida rippmenüüst **Tools**.
3. Seejärel **Block Import and Export Wizard**....(Joonis 12)
4. Järgnevalt **Browse** ning leida plokki asukoht. (Joonis 13)
5. Valida **Palette**, kuhu plokk paigaldada.
6. Seejärel vajutada **Import**.



Joonis 16. NXT-G rippmenüü.



Joonis 17. NXT-G plokki importimine.

## LISA 4 – Ploki kasutusjuhend



### BTMemory plokk

Vastav blokk saadab mälumoodulile käsu faili avamiseks, kirjutamiseks ning sulgemiseks.

Ploki funktsiooni saab muuta konfiguratsiooni menüüs.

**Avamise funktsiooni** kasutatakse faili avamiseks või loomiseks.

**Kirjutamise funktsiooni** kasutatakse faili kirjutamiseks.

**Sulgemise funktsiooni** kasutatakse faili sulgemiseks.

Faili kirjutamiseks peab esmalt avama faili, seejärel kirjutama andmed sisse ning seejärel tuleb fail sulgeda.

### BTMemory ploki diagramm



#### Plokk

1. **Text** - lisatakse kogutud andmetele lisainformatsioonina.
2. **Reading** - kogutud andmed või defineeritud andmed.

#### Avatud plokk



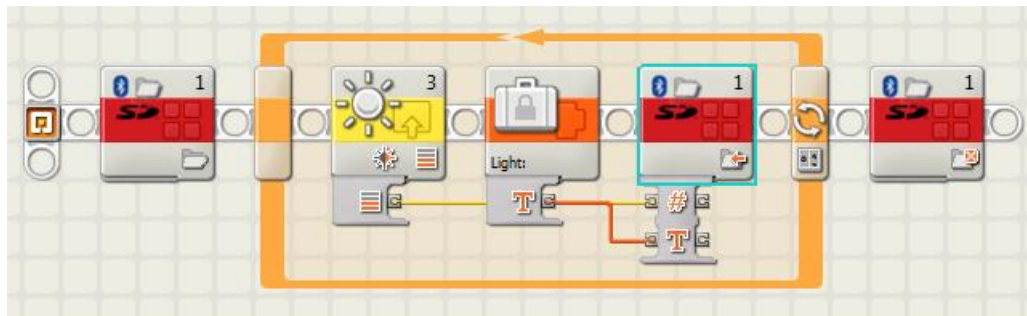
1. **Indicator** - näitab millisesse ühendusse on Bluetooth ühendus loodud.
2. **Indicator** - näitab, mis funktsioon on kasutusel: Avamine (Open), Kirjutamine (Write) või Sulgemine (Close).
3. **Wait** - ajaintervall kogutavate andmete vahel.
4. **Number** - võimalus lubada/keelata numbrit.
5. **Time** - võimalus lubada/keelata ajatemplit.
6. **Filename** - failinimi.
7. **Text** - koguvate andmete lisainformatsioon.
8. **Reading** - kogutavad numbrilised andmed või defineeritud nukbrilised andmed.
9. **Command** - Avamine (Open), Kirjutamine (Write) või Sulgemine (Close).
10. **BT** - Bluetooth ühenduse number.

## BTMemory konfiguratsiooni paneel

<b>BTMemory Module Block</b> 	Command:	1	Open	Text:	5	
	Bluetooth connection:	2	Connection 1	Number	6	<input checked="" type="checkbox"/>
	File name:	3		Reading:	7	0
	Time	4	<input checked="" type="checkbox"/>	Wait [s]:	8	1

1. **Command:** - Avamine (Open), Kirjutamine (Write) või Sulgemine (Close).
2. **Bluetooth connection:** - 1-3 Bluetooth ühendus.
3. **File name:** - Faili nimi, kuhu andmed salvestatakse.
4. **Time:** - Lisa ajatempel.
5. **Text:** - Lisainformatsiooni tekst.
6. **Number:** - Numbri lubamine/keelamine.
7. **Reading:** - Kogutav numbriline väärtus.
8. **Wait [s]:** - Ajaintervall kogutavate andmete vahel sekundites.

## Näidisprogramm



Vastav näidisprogramm loob faili, kirjutab valgusandurilt saadud andmed SD kaardile koos tekstiga "Light: " teatud aja jooksul ja sulgeb faili.

**Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks**

Mina \_\_\_\_\_ Kait Krull \_\_\_\_\_  
(*autori nimi*)  
(sünnikuupäev: \_\_\_\_\_ 16.11.1991 \_\_\_\_\_)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

\_\_\_\_\_  
“Lisamälu ühendumisvõimaluse loomine MINDSTORMS platvormile“ \_\_\_\_\_,  
(*lõputöö pealkiri*)

mille juhendaja on \_\_\_\_\_ Heilo Altin \_\_\_\_\_,  
(*juhendaja nimi*)

- 1.1.reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
  - 1.2.üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
  3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus/Tallinnas/Narvas/Pärnus/Viljandis, 20.05.2013 (*kuupäev*)